CLAIMS

What is claimed is:

1. A computer system comprising:

a plurality of microprocessors, each microprocessor having a cache;

a main memory array, a portion of the main memory array designated as a first-in/first-out (FIFO) buffer;

a first bridge device coupling the plurality of microprocessors and the main memory array, the first bridge device at least partially responsible for implementing a cache coherency protocol to keep the cache of each microprocessor and the main memory coherent;

a second bridge device coupled to the first bridge device by way of a primary expansion bus;

a hardware device coupled to the second bridge device by a secondary expansion bus, wherein the hardware device has a cache memory that duplicates a portion of the FIFO buffer, and wherein the hardware device cache memory is kept coherent by way of the cache coherency protocol.

2. The computer system as defined in claim 1 further comprising:

at least one of the plurality of microprocessors executing a software stream; and

said software stream configured to pass bytes of information to the hardware device by only placing the bytes of information in the FIFO buffer.

3. The computer system as defined in claim 2 further comprising said hardware device adapted to poll the cache memory that duplicates portions of the FIFO to check for availability of bytes of information from the software stream.

4. The computer system as defined in claim 1 wherein the FIFO buffer of the main memory array further comprises a set of continuously addressed memory locations.

5. The computer system as defined in claim 4 wherein the FIFO buffer further comprises at least one cache line of memory locations.

6. The computer system as defined in claim 5 wherein the cache line of memory locations is 128 bytes in length.

7. The computer system as defined in claim 1 wherein the first bridge device further comprises:

a first register identifying a beginning location of the FIFO buffer that is duplicated by the cache memory of the hardware device;

a second register identifying an end location of the FIFO buffer that is duplicated by the cache memory of the hardware device;

a destination register identifying a location of the hardware device; and

wherein the first bridge logic, as part of the cache coherency protocol, compares transactions to addresses in main memory to the first register and the second register to determine

if the transaction is directed to a memory location duplicated by the onboard cache memory of the hardware device.

8. The computer system as defined in claim 7 wherein the cache coherency protocol further comprises a write-back invalidate cache protocol.

9. The computer system as defined in claim 7 wherein the first register contains an address of a first memory location of the FIFO buffer.

10. The computer system as defined in claim 9 wherein the second register contains an address of a last memory location of the FIFO buffer.

11. The computer system as defined in claim 9 wherein the second register contains an offset representing the number of memory locations a last address of the FIFO buffer resides from the first memory address.

12. The computer system as defined in claim 1 wherein the second bridge device further comprises:

a first register identifying a first cached memory address;

a second register identifying a second cached memory address;

a third register identifying the hardware device;

wherein the first and second registers identify a series of continuous memory addresses of the main memory cached by the hardware device identified in the third register; and

wherein the second bridge device receives cache coherency protocol messages, compares addresses of the cache coherency protocol messages to the first and second registers, and forwards the messages to the device identified in the third register.

13. The computer system as defined in claim 12 wherein the first register contains an address of a first memory location of the FIFO buffer.

14. The computer system as defined in claim 13 wherein the second register contains an address of a last memory location of the FIFO buffer.

15. The computer system as defined in claim 13 wherein the second register contains an offset representing the number of memory locations a last address of the FIFO buffer resides from the first memory address.

16. The computer system as defined in claim 1 wherein the hardware device coupled to the second bridge device by the secondary expansion bus further comprises a hardware device capable of bus-mastering the secondary expansion bus.

17. The computer system as defined in claim 16 wherein the hardware device further comprises a network interface card.

18. The computer system as defined in claim 17 wherein the network interface card further comprises a system area network interface card.

19. The computer system as defined in claim 18 wherein the system area network interface card further comprises an Infini Band compatible interface card.

20. The computer system as defined in claim 17 wherein the network interface card further comprises a storage area network interface card.

21. The computer system as defined in claim 20 wherein the network interface card further comprises a Fibre Channel compatible interface card.

22. The computer system as defined in claim 16 wherein the hardware device further comprises a graphics adapter.

23. The computer system as defined in claim 16 wherein the hardware device further comprises an audio input/output card.

24. The computer system as defined in claim 16 wherein the hardware device further comprises a mass storage device.

25. The computer system as defined in claim 24 wherein the mass storage device further comprises a hard drive.

26. The computer system as defined in claim 24 wherein the mass storage device further comprises a compact disk drive.

27. The computer system as defined in claim 1 wherein the cache memory that duplicates a portion of the FIFO memory in the hardware device further comprises a random access memory configured to operate as the cache memory.

28. The computer system as defined in claim 1 wherein the cache memory that duplicates a portion of the FIFO memory in the hardware device further comprises a series of hardware registers.

29. A method of notifying a hardware device in a computer system that information from a software stream executed by a microprocessor is available in a main memory array first-in/first-out (FIFO) buffer, the method comprising:

allowing the hardware device to participate in a coherency domain of the computer system by the hardware device having a coherent cache memory duplicating a cache line of the FIFO buffer;

writing information to the cache line of the FIFO buffer by the software stream; and

notifying the hardware device that the information is available in the FIFO buffer by invalidating the duplicate copy of the cache line of the FIFO buffer in the onboard cache memory of the hardware device.

30. The method as defined in claim 29 wherein writing information to the cache line of the FIFO buffer by the software stream further comprises:

requesting exclusive ownership of the cache line by a microprocessor executing the software stream;

granting exclusive ownership of the cache line to the microprocessor by a cache coherency system; and

writing the cache line once exclusive ownership of the cache line is granted to the microprocessor.

31. The method as defined in claim 30 wherein notifying the hardware device that the information is available in the FIFO buffer further comprises invalidating the duplicate copy of the cache line in the cache memory of the hardware device substantially simultaneously with the granting exclusive ownership step.

32. The method as defined in claim 31 wherein invalidating the duplicate copy of the cache line further comprises sending an invalidation message from the cache coherency system to the hardware device to invalidate the copy of the cache line in the cache memory of the hardware device.

33. The method as defined in claim 29 further comprising obtaining a copy of the cache line by the hardware device after receiving the invalidation command.

34. The method as defined in claim 33 wherein obtaining a copy of the cache line by the hardware device further comprises:

arbitrating by the hardware device for mastership of a secondary expansion bus; and

reading the cache line of the FIFO buffer into the cache memory.

35. The method as defined in claim 29 further comprising transferring response information from the hardware device to the software stream by the hardware device writing the response information to a second first-in/first-out (FIFO) buffer in the main memory array.

36. The method as defined in claim 29 wherein notifying the hardware device that the information is available further comprises:

polling by the hardware device of the cache memory; and

receiving notification that the information is available in the FIFO buffer based on the hardware device sensing that the cache line of the FIFO buffer in the cache is invalid.

37. A method of transferring data between a software program executed on a microprocessor and a hardware device coupled to an expansion bus in a computer system, the method comprising:

providing an exchange buffer in main memory;

providing a duplicate copy of at least part of the exchange buffer in the hardware device;

writing data by the software program into the exchange buffer, making the data in the exchange buffer different than the duplicate copy of the exchange buffer in the hardware device;

invalidating the duplicate copy of the at least part of the exchange buffer in the hardware device; and

reading the exchange buffer contents by the hardware device responsive to the invalidating step.

38. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 37 wherein providing an exchange buffer in main memory further comprises designating a portion of the main memory as a first-in/first-out (FIFO) buffer.

39. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 38 wherein designating a portion of the main memory as a FIFO buffer further comprises designating a set of continuous memory addresses as the FIFO buffer.

40. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 37 wherein providing a duplicate copy of at least part of the exchange buffer in the hardware device further comprises providing a cache memory local to the hardware device that duplicates the at least part of the exchange buffer.

41. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 37 wherein writing data by the software program into the exchange buffer further comprises:

copying data from the exchange buffer into a cache of the microprocessor;

writing data to the copy of the exchange buffer in the cache of the microprocessor by the software program; and

writing the data from the microprocessor cache to the exchange buffer in main memory consistent with a cache coherency protocol.

42. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 41 wherein writing the data from the microprocessor cache to the exchange buffer in main memory consistent with a cache coherency protocol further comprises writing the data from the microprocessor cache to the exchange buffer consistent with a write-back cache coherency protocol.

43. The method of transferring data between a software program and a hardware device in a computer system as defined in claim 41 wherein writing the data from the microprocessor cache to the exchange buffer in main memory consistent with a cache coherency protocol further comprises writing the data from the microprocessor cache to the exchange buffer consistent with a write-through cache coherency protocol.

44. A computer system comprising:

a microprocessor means for executing software programs, the microprocessor means having a cache means;

a main memory means for providing program and data storage, a portion of the main memory means designated a buffer means for providing data exchange;

a first bridge means for coupling the microprocessor means and the main memory means, the first bridge means also for implementing at least a portion of a cache coherency protocol to keep the cache means and the main memory means coherent;

a second bridge means for providing protocol translation between a first communication bus means and a second communication bus means, the first communication bus means for coupling the first bridge means to the second bridge means; and

a hardware device means for providing hardware specific tasks, the hardware device means coupled to the second bridge means by the second communication bus means, wherein the hardware device means has an cache memory means for duplicating a portion of the buffers means, and wherein the cache memory means is kept coherent by way of the cache coherency protocol.

45. The computer system as defined in claim 44 wherein the buffer means further comprises a first-in/first-out (FIFO) buffer.

46. The computer system as defined in claim 44 wherein the FIFO buffer of the main memory means further comprises a set of continuously addressed memory locations in the main memory means.

47. The computer system as defined in claim 46 wherein the FIFO buffer further comprises at least one cache line of memory locations.

48. The computer system as defined in claim 47 wherein the cache line of memory locations is 2^N bytes in length, where N is an integer.

49. The computer system as defined in claim 44 wherein the first bridge means further comprises:

a first register means for storing an indication of a beginning location of the buffer means that is duplicated in the cache means of the hardware device means;

a second register means for storing an indication of an end location of the buffer means that is duplicated in the cache means of the hardware device means;

a destination register means for storing an address identifying a location of the hardware device means; and

wherein the first bridge means, as part of the cache coherency protocol, compares transactions to addresses in the main memory means to the first register means and the second register means to determine if the transaction is directed to a memory location cached by the hardware device means.

50. The computer system as defined in claim 49 wherein the cache coherency protocol further comprises a write-back invalidate cache protocol.

51. The computer system as defined in claim 49 wherein the first register means contains an address of a first memory location of the buffer means.

52. The computer system as defined in claim 51 wherein the second register means contains an address of a last memory location of the buffer means.

53. The computer system as defined in claim 51 wherein the second register means contains an offset representing the number of memory locations a last address of the buffer means resides from the first memory address.

54. The computer system as defined in claim 44 wherein the second bridge means further comprises:

a first register means for storing a value identifying a first cached memory address;

a second register means for storing a value identifying a second cached memory address;

a third register means identifying the hardware device means;

wherein the first and second register means identify a series of continuous memory addresses of the main memory means duplicated by the hardware device means; and

wherein the second bridge means receives cache coherency protocol messages, compares addresses of the cache coherency protocol messages to the values in the first and second registers means, and forwards the messages to the device identified in the third register if the addresses fall within the values.

55. The computer system as defined in claim 54 wherein the first register means contains an address of a first memory location of the buffer means.

56. The computer system as defined in claim 55 wherein the second register means contains an address of a last memory location of the buffer means.

57. The computer system as defined in claim 55 wherein the second register means contains an offset representing the number of memory locations a last address of the buffer means resides from the first memory address.

58. The computer system as defined in claim 44 wherein the software stream executed by the microprocessor means is configured to pass bytes of information to the hardware device means by only by placing the bytes of information in the buffer means.

59. The computer system as defined in claim 45 further comprising said hardware device means adapted to poll the cache means that duplicates the portion of the buffer means to check for availability of bytes of information from the software stream.

60. The computer system as defined in claim 44 wherein the hardware device means coupled to the second bridge means by the second communication bus means further comprises a hardware device capable of bus-mastering the second communication bus means.

61. The computer system as defined in claim 60 wherein the hardware device further comprises a network interface card.

62. The computer system as defined in claim 61 wherein the network interface card further comprises a system area network interface card.

63. The computer system as defined in claim 62 wherein the system area network interface card further comprises an Infini Band compatible device.

64. The computer system as defined in claim 61 wherein the network interface card further comprises a storage area network interface card.

65. The computer system as defined in claim 64 wherein the storage area network interface card further comprises a Fibre Channel compatible device.

66. The computer system as defined in claim 60 wherein the hardware device further comprises a graphics adapter.

67. The computer system as defined in claim 60 wherein the hardware device further comprises an audio input/output card.

68. The computer system as defined in claim 60 wherein the hardware device further comprises a mass storage device.

69. The computer system as defined in claim 68 the mass storage device further comprises a hard drive.

70. The computer system as defined in claim 68 the mass storage device further comprises a compact disk drive.

71. The computer system as defined in claim 44 wherein the cache memory means for duplicating a portion of the buffer means in the hardware device means further comprises a random access memory configured to operate as a cache memory.

72. The computer system as defined in claim 44 wherein the cache memory means for duplicating a portion of the buffer means in the hardware device means further comprises a series of hardware registers configured to operate as a cache memory.